Towards Trust Engineering for Opportunistic Cloud Services

A Systematic Review of Trust Engineering in Cloud Computing

Kuada, Eric

Publication date:
2014

Document Version
Accepted author manuscript, peer reviewed version

Towards Trust Engineering for Opportunistic Cloud Services:
A Systematic Review of Trust Engineering in Cloud Computing

Eric Kuada
Department of Electronic Systems
Aalborg University, Copenhagen, Denmark
kuada@cmi.aau.dk

Abstract -The systematic review methodology has been employed to review trust related studies in cloud computing. It was observed that trusted computing technologies and reputation based approaches are the main approaches to trust engineering in cloud computing. Also, trusted third party approaches and the deployment model play a significant role in enhancing trust between service providers and consumers. It was observed that the concept of trust is used loosely without any formal specification in cloud computing discussions and trust engineering in general. As a first step towards addressing this problem, we have contextualized the formal trust specification in multi-agent environments for cloud computing.

Keywords- security engineering; trust engineering; trust in cloud computing; trust modeling.

1 INTRODUCTION

This introductory section begins with the motivation for undertaking this study. Next, a background to the need for trust engineering in cloud computing is provided. An overview to opportunistic cloud services (which is the foundation for the motivation of this study) is also given. Since the methodological approach to this study is systematic literature review, the section ends with a brief introduction to systematic literature reviews and the processes involved in conducting such a review.

1.1 Motivation

We have over the past three years been working on the feasibility of Opportunistic Cloud Services (OCS) for enterprises[1] [2]. One of the major challenges that such a platform faces is data security and trust management issues. In order to design and develop a trust management system for OCS platforms, we needed to review the current trust engineering issues in cloud computing. It was decided that we needed to perform a systematic literature review on this topic because since the OCS concept is itself new, any trust design models of its subsystems must be guided by exhaustive knowledge of the state-of-the-art in the field. The rigorous methodological approach offered by systematic literature reviews will ensure that the topic is adequately covered. The objective of this paper is therefore to provide state-of-the-art knowledge on trust engineering concepts and models in cloud computing.

1.2 Background to Trust in Cloud Computing

Cloud computing is essentially the packaging of traditional Information Technology infrastructure and software solutions such as storage, CPU, network, applications, services, etc., as virtualized resources and delivered by a service provider to its customers as an on-demand pay-per-use self-provisioned service, which is normally offered through a web portal over a network such as the Internet[3] [4] [5]. While cloud service providers pledge to preserve data privacy, the current Software as a Service (SaaS) architecture makes it difficult to provide any assurance that the software in the Cloud will not be able to make copies or redistribute the data it used[6] . Secondly, the Cloud model is based on two key characteristics: multi-tenancy, where multiple tenants share the same service instance, and elasticity, where tenants can scale the amount of their allocated resources based on current demands. Although both characteristics target improving resource utilization, cost reduction, and service availability, these gains are threatened by multi-tenancy security implications. The sharing of applications that process critical information without sufficient proven security isolation, security SLAs or tenant control, results in “loss-of-control” and “lack-of-trust” problems[7].

Apart from these consumer concerns, cloud architectures also introduce new classes of security risks and attacks over the resources of cloud service providers. These include poisoned virtual machines, attacks against the cloud service provider’s management console, attacks based on knowledge of default security settings, abuse of billing systems, and data leakage via uniform resource locators. Cloud service providers still do not currently have sufficiently robust technical solutions that can

protect their cloud resources from harmful malware, virus infection, botnets, distributed denial of service attacks, or other types of cyber-attacks. Furthermore, there is no effective mechanism to help cloud users evaluate the security measures of their service providers and ensure the protection of their data while taking into consideration industry standards or personal preferences [8].

1.3 Opportunistic Cloud Services

Opportunistic Cloud Services (OCS) is a social network approach to the provisioning and management of cloud computing services for enterprises. OCS is about enterprises leveraging free cloud services to meet their business needs without having to pay or paying a minimal fee for these services [9][10]. An OCS network is a social network of enterprises collaborating strategically for the contribution and usage of cloud services without entering into any business agreements[1]. Members normally will package only their spare IT resources and make them available as Cloud services on the OCS platform so that others interested can utilize them. Since no business agreement and hence no Service Level Agreement (SLA) exist between the service providers and the potential users of their services, service consumers do not enjoy the level of support (in terms of quality of service, reliability, availability, security, billing transparency, etc.) that commercial cloud service providers offer to their clients. Considering the fact that commercial cloud service providers are finding it extremely challenging to provide such a support, coupled with having to provide adequate transparency in their management processes, the OCS platform more so needs a well-crafted and soundly engineered trust management system in order to make resources on the platform suitable for business use.

1.4 Systematic Literature Reviews

A systematic literature review is a means of identifying, evaluating and interpreting all available research - that are known to the researcher - and relevant to a particular research question, topic area, or phenomenon of interest [11][12]. It is a systematic, explicit, comprehensive, and reproducible method for identifying, evaluating, and synthesizing the existing body of completed and recorded work produced by researchers, scholars, and practitioners on a specific topic of interest [13]. The accumulation of evidence through secondary studies can be very valuable in offering new insights or in identifying where an issue might be clarified by additional primary studies. The systematic literature review process consists of three main stages - namely inputs, processing, and outputs [14]. The eight step guide of [13] to conducting systematic literature review are: purpose of the literature review, protocol and training, searching for the literature, practical screen, quality appraisal, data extraction, synthesis of studies, and finally writing the review. They recommend all these steps are essential for a review to be scientifically rigorous. According to[11] the stages in a systematic literature review can be summarized into three main phases: planning the review, conducting the review, and reporting the review. The stages associated with *planning the review* are: identification of the need for a review, specifying the research questions, and developing a review protocol; the stages associated with *conducting the review* are: identification of relevant existing research, selection of primary studies, study quality assessment, data extraction and monitoring, and data synthesis; and finally, the stages associated with *reporting the review* are: specifying dissemination mechanisms, and formatting the main report.

2 METHODOLOGY

We adopt a blend of the guidelines of [11] and [13] because after a careful analysis, we consider both guides to be suitable for our purpose; and it was evident that their main individual stages are in agreement and refer to the same concepts with slightly different tagging.

2.1 Planning the Review

The main activities involved in planning the review are specifying the objectives of the study, specifying the research questions, developing and evaluating the review protocol, and justifying the need for the study.

2.1.1 Need for the Study

There have been efforts on surveys on security issues in cloud computing [15] [16] but not on trust engineering. Also, even though security is a key element of trust, these studies are not systematic reviews and those that attempt a systematic review such as [17] focus only on security; so to the best of our knowledge, this study is the first attempt of summarizing the body of knowledge on trust engineering in cloud computing environments.

2.1.2 Objectives and Research Questions

The first of the main objectives of this study is to provide state-of-the-art knowledge on trust engineering concepts and models in cloud computing. The second objective is to provide a firm grounding for engineering a trust model and trust management system for opportunistic cloud computing services. Based on these objectives, the research questions that are of interest to this study are:

a. What are the main approaches towards trust engineering in cloud computing?
b. What are the major trust models and trust management systems for cloud computing?
c. What are the objectives of the identified primary studies and in what contexts are these trust management systems being developed?

2.1.3 Review Protocol

A review protocol specifies the methods that will be used to undertake a specific systematic review. The components of a protocol include all the elements of the review plus some additional planning information such as the project timeline [11]**.** The entire methodology section in this paper gives a summary of review protocol that has been applied in undertaking this study. The review protocol has been under constant re-evaluation to ensure that the applied search strings are derived from the research questions; the extracted data properly address the research questions; and the data analysis procedure is appropriate to answer the research questions.

2.2 Conducting the Review

The stages associated with conducting the review are identification and selection of relevant existing primary studies, study quality assessment, data extraction and monitoring, and data synthesis**.**

2.2.1 Search Strategy

The adopted search strategy is to search for keywords in standard metadata (i.e. title, abstract, and author keywords). The keywords derived from our topic "Trust Engineering in Cloud Computing" are trust, engineering, and cloud computing. However, because privacy and security are two major elements in trust in cloud computing, we expanded our search keywords to include them. Additionally, to ensure the search strings are derived from our research questions, we further expand the keywords to include model. We then use a combination of two or more of the resulting keywords as search strings in searching for the primary resources for this study. The resulting search strings are: trust engineering, trust cloud computing, trust model, security in cloud computing, privacy in cloud computing, security engineering, and privacy engineering.

2.2.2 Sources

Considering the topic of this study, the major sources that the search strategy was applied in are IEEE Xplore Digital Library, ACM Digital Library, Google scholar, and Journals of Elsevier and Springer Link. These sources were supplemented with the general Internet and the Aalborg University digital library portal, Primo, which is a portal into well-known research databases.

2.2.3 Practical Screen and Quality Appraisal

Refworks [18] was used as the bibliographic management tool for managing the large number of over five hundred studies resulting from the search process. These were taken through practical screening by reading through their abstracts and those that didn't have relation to our research topic were excluded; leaving about 320 primary studies as our base resources. A second and a third round of reviews were performed to select those that had direct bearing on the research questions. This process yielded about 140 articles that have been included in this study. All these articles were then retrieved and processed for the data extraction phase. This process spanned a period of four months, from September to December 2012. Regular update to the list of articles was made during data extraction and synthesis of studies phases in the subsequent months of undertaking this study.

2.2.4 Data Extraction

NVivo10[19] was the choice of tool for the data extraction phase. Even though we are well aware of other qualitative data analysis tools such as Atlas.ti [20], we did not consider them since the university had license to only NVivo. All these relevant primary studies were manually read. The basic methodological steps of constant comparison for coding in grounded theory [21] [22]were applied in the data extraction process with help of Nvivo in the coding of the data as we read through each article. Furthermore the grounded theory methodology allowed extraction of relevant information (e.g. the major challenges of trust engineering in cloud computing) from the primary studies, even though these were not initially part of the focus of our study and hence did not reflect in our research questions.

2.2.5 Synthesis of Studies

During the synthesis stage, major trends that had developed during the coding process were further investigated by searching for new articles on these specific topics in order to shed more light on them. We followed an iterative process of categorization and reorganization of the extracted data, supplemented with finding new articles that support or weaken the trends being observed. Those categories lacking adequate support and could also not fit naturally under other categories did not merit further analysis and were dropped in our discussions as is presented below.

3 ANALYSIS & DISCUSSION

We have followed an iterative process of categorization and reorganization of the extracted data, supplemented with finding new articles that support or weaken the trends being observed, in order to obtain the final headings that are discussed in this section. The main areas covered in our analyses and discussions are:

- Trust, security and privacy challenges in cloud computing
- Focus on trust engineering in cloud computing
- Modeling trust: this deals with the modeling of the concept of trust.
- Trust engineering approaches
- Trust management systems

3.1 Trust, Security and Privacy Challenges in Cloud Computing

Though the identification of challenges in cloud computing was not part of the research objectives or questions that were spelt out during the "Planning the Review" stage, it was evident during the data extraction process that it is a paramount issue that needed some attention. The major challenges in cloud computing as were reported by the reviewed papers can be categorized into trust challenges, security challenges, and privacy challenges. This categorization however does not mean the categories are mutually exclusive, as it will be seen later that for example, security and privacy issues impact upon the perceived trust of various entities in a cloud computing marketplace.

3.1.1 Trust Challenges

An important issue in cloud computing is the accountability of the resource usage data: who performs the measurement to collect resource usage data – is it the provider, the consumer, a trusted third party or some combination of them? Currently, provider side accountability is the basis for cloud service providers, although, as yet there are no equivalent facilities of consumer-trusted metering as is the case in traditional utility services; rather, consumers have no choice but to take whatever usage data made available by the provider as trustworthy [23].

Another issue concerning trust in cloud computing is that, potential customers of cloud services often feel that they lose the control over their data, and they are not sure whether they can trust the cloud service providers. A survey conducted in 2011 among more than three thousand cloud consumers from six countries, shows that 84 percent of the consumers are concerned about their data storage location and 88 percent of the consumers worry about who has access to their data. Though consumer concerns can be mitigated by using preventive measures for privacy (e.g., demonstrating compliance standards) and security (e.g., secure hypervisors, TPM based servers), at present, cloud providers demonstrate their preventive measures by including related descriptions in the SLAs; assurances and compensations for SLA violations are however not convincing enough for the consumers. Especially, SLAs with vague clauses and unclear technical specifications lead the consumers into a decision dilemma when considering them as the only bases to identify trustworthy providers [24].

A third issue concerning trust in cloud computing is that, the SaaS model gives software providers an unprecedented access to data uploaded by users. At execution time the control of the data is handed over from the user (data owner) to the software provider. Furthermore, the results generated from the software execution, in theory, are under the control of the software provider. This raises a new concern about trust on software providers [6]. Data must be decrypted into memory when performing the computation, even though they can be encrypted during storage and transmission. In this case, the privileged administrators of SaaS providers are able to inspect or modify users' data and computations. As a result, the users are hesitant to trust the SaaS providers [25].

3.1.2 Security Challenges

Possible misuse of customers' data by cloud service providers is a major challenge in cloud computing. The privileged administrators of cloud service providers are able to inspect, modify, or misapply users' data and computations. Some of the security challenges facing cloud computing are multi-tenancy security implications, security isolation, cloud service providers' and customers' need of modeling and enforcing different security requirements (especially at runtime because security requirements may change over time as new risks emerge), and integrating with different security services. After analyzing the cloud computing model security problem, and information security management systems (ISMS) process, [26] has identified the following key problems:

- Each stakeholder has their own security management process (SMP) that they want to maintain or extend to the cloud hosted assets.
- No stakeholder can individually maintain the whole security process of the cloud services because none of them has the full information required to manage security and each one has a different perspective.
- Multi-tenancy requires maintaining different security profiles for each tenant on the same service instance.
- No Security SLA is available that can be used to maintain agreements related to cloud assets security.
- The existing standards such as ISO27000 and FISMA do not map well to the cloud model because these standards consider the SMP from the perspective of the platform/asset owner, not from a service provider perspective.

While there might be a multitude of operating systems (OSs) deployed in a single cloud, the majority of such OSs have not been designed for the Cloud. In particular, traditional logging is process and/or event-based (for a particular user or node). In the Cloud, however, there are no clear user or node barriers; instead, logging should be done with respect to the key assets, i.e., data and information. In terms of OSs, this means data-centric logging. Besides provenance, other key concerns mandating data- centric logging include the need for support of consistency assurance, rollback, recovery, replay, backup, and restoring of data. Such functionality is usually enabled by using operational and/or

transactional logs. Such logs have also been proven useful for monitoring of operational anomalies. While these concepts are well established in the database domain, cloud computing's characteristics such as eventual consistency, 'unlimited' scale, and multi-tenancy pose new challenges. In addition, secure and privacy-aware mechanisms must be devised not only for consistency logs but also for their backups, which are commonly used for media/node recovery [27].

Data processing clouds, including Hadoop[28], execute untrusted, user-submitted code on trusted cloud nodes during job processing, and must therefore remain vigilant against malicious mobile code attacks. Virtualization technologies, including trusted hardware, hypervisors, secure operating systems, and trusted VMs are the typical means by which such mobile code is secured. However, a variety of studies have shown that clouds introduce significant new security challenges that make mobile code security a non-trivial, ongoing battle. For example, the Cloud Security Alliance has identified insecure cloud APIs, malicious insiders, shared technology issues, service hijacking, and unknown risk profiles all as top security threats to cloud services [29].

Adopting multi-tenancy with SaaS results in a set of requirements that must be addressed by the SaaS application. Two key requirements in the area of SaaS applications' security engineering have been identified by [7]. The first one is the security isolation among tenants' assets at rest (storage), during processing (in memory), and during transient (among application components or between the application and the tenant site). Secondly, it is required to support enforcement of different security requirements on the same service instance at runtime. Application customization approaches do not fit well with runtime and multi-tenant specification and security enforcement because these security requirements may change over time as new risks emerge.

Data integrity is another major security challenge for cloud computing. It is most often assumed that the underlying storage arrays (similar technologies of which are being employed by cloud service providers), receive, store and retrieve data flawlessly. This assumption is however proven to be false in the past, as evident from the CERN report[30] and other studies[31]. Therefore, prompt detection of integrity violations is vital for the reliability and safety of the stored data in the Cloud [32].

3.1.3 Privacy Challenges

In cloud computing, entities may have multiple accounts associated with a single or multiple service providers (SPs). Sharing sensitive identity information (i.e. Personally Identifiable Information (PII)) along with associated attributes of the same entity across services can lead to mapping of the identities to the entity; and this is leads to privacy loss. The major problems regarding privacy in the Cloud include how to secure PII from being used by unauthorized users; how to prevent attacks against privacy (such as identity theft) even when a cloud SP cannot be trusted; and how to maintain control over the disclosure of private information [33].

As has been indicated by [34], there are situations where cloud service providers themselves invade the privacy of their users, so a cloud service provider is generally not the entity to fully rely on in order to protect one's privacy. Consequently, there is a need for additional external measures to protect a user's privacy. This need has been recognized in several previous approaches for protecting data in the Cloud [35] [36]. However, these approaches suffer from bad usability and require too much effort from the users, as shown for example by Whitten and Tygar [37] and subsequent user tests. There are theoretically many cryptographic mechanisms that would perfectly suit the privacy needs of today's Internet users, but their use is avoided due to a lack of good usability and high effort required. For example, Public Key Infrastructures (PKIs) burden the users with handling cryptographic artifacts. Although there are many efforts to simplify the usage of a PKI ,e. g. [38] [39], the majority of users still shy away from the extra work [34].

3.2 Cloud Computing Trust Engineering Focus

We now analyze the main objectives of researchers on trust engineering in cloud computing to determine what trust engineering research has focused on within the past few years. We extract the objectives of selected works of which the objectives had been clearly stated (normally stated in the abstract or in the introductory sections), or can be easily inferred from these sections. We have identified five main research focuses on trust engineering in cloud computing. They are performance and Quality of Service (QoS), security related, access and Identity management, user and provider support on trust management, and billing and accountability. We end the section with some concluding remarks on some of the salient points of these research areas together with the context within which these studies had been carried out.

3.2.1 Performance and QoS

The objective of the trust evaluation model of [40] is to configure the complex set of services dynamically in a cloud environment according to the predictive performance in terms of stability and availability of all services that are to be provided; this is with the aim of allowing a system to configure services dynamically and distribute tasks efficiently in such a way that minimizes task failure and task migration rate.

Success of cloud computing requires that both customers and providers can be confident that signed SLAs are supporting their respective business activities to their best extent. The SLAs currently being used fail in providing such confidence, especially when providers outsource resources to other providers. These resource providers typically support very simple metrics like

availability, or metrics that hinder an efficient exploitation of their resources. A resource-level metric for specifying fine-grain guarantees on CPU performance has been proposed by [41].

Due to the dynamic nature of cloud computing, how to achieve satisfactory QoS in cloud workflow systems becomes a challenge. Meanwhile, since QoS requirements have many dimensions, a unified system design for different QoS management components is required to reduce the system complexity and software development cost;[42] has therefore proposed a generic QoS framework for cloud workflow systems. The framework covers the major stages of a workflow lifecycle. It consists of QoS requirement specification, QoS-aware service selection, QoS consistency monitoring and QoS violation handling.

3.2.2 Security

The aim of [43] is to provide a system that makes it possible to detect that at least the configuration of the cloud infrastructure -as provided in the form of a hypervisor and administrative domain software- has not been changed without the customer's consent. They present a system that enables periodical and necessity-driven integrity measurements and remote attestations of vital parts of cloud computing infrastructures. The objective of [43] is to tackle the problem of protecting entities using the Cloud from malicious or negligent entities providing the cloud infrastructure. They present the *BonaFides* system for remote attestations of security-relevant parts of the cloud infrastructure, which guarantees to service providers at runtime the detection of unintended or malicious modifications of cloud infrastructure configurations. Their approach does not prevent the cloud infrastructure provider from altering crucial components and subsequently stealing data, but these activities will at least be detected by the cloud consumers.

The objectives of [25] is to provide a trusted SaaS platform (TSP) which will guarantee data security during storage and transmission, and also enforce a trusted execution environment (TEE) that guarantees the confidentiality and integrity of the users' data and computations. The objective of [7] is to provide a security management architecture- Tenant Oriented SaaS Security Management Architecture (TOSSMA) - that allows service providers to enable their tenants in defining, customizing and enforcing their security requirements without having to go back to application developers for maintenance or security. The objective of [32] is to offer a secure cloud storage service architecture with the focus on Data Integrity as a Service (DIaaS) based on the principles of Service-Oriented Architecture and Web services. The approach releases the burdens of data integrity management from a storage service by handling it through an independent third party data Integrity Management Service (IMS); it also reduces the security risk of the data stored in the storage services by checking the data integrity with the help of IMS.

In order to address privacy and security issues, and to incorporate security and trust functionalities that complies with EU and government privacy laws, [44] has presented the Cloud Data Security (CloudDataSec) project that aims to design cloud services adhering to government privacy laws. In particular, they introduced a six-layer security model for cloud computing and three level of security assurance for SMEs to take advantage of. Finally, they proposed Security Management as a Service (SMaaS) modules to enable users to apply necessary security and privacy operations, based on the sensitivity of their data.

The objective of [26] is to introduce a cloud security management framework based on aligning the FISMA standard[45][46] to fit with the cloud computing model; this is with the aim of enabling cloud providers and consumers to be security certified through improving collaboration between cloud infrastructure providers, cloud service providers and service consumers in managing the security of the cloud platform and the hosted services.

3.2.3 Access and Identity Management

Because available solutions to identity management in cloud computing use trusted third party (TTP) in identifying entities to service providers, and these solution providers do not recommend the usage of their solutions on untrusted hosts, the objective of [33] is to develop a framework for identity management which is independent of TTP and has the ability to use identity data on untrusted hosts. The objective of [47] is to provide a mechanism (Trust Ticket) of ensuring trust and security in Software as a Service (SaaS). Their Trust Ticket, together with the supporting protocols, is a mechanism that helps a data owner in establishing a link between a cloud service provider and a registered user. In this mechanism, a user first gets registered with a data owner before receiving a Trust Ticket and a secret key from that data owner. Each Trust Ticket is unique and encrypted. On completing the registration of each user, the data owner apprises the cloud service provider of the Trust Ticket.

3.2.4 User and Provider Trust Management Support

Due to the vast diversity in the available cloud services, from the customers' point of view, it has become difficult to decide whose services they should use and what the basis for their selection is. Currently, there is no framework that can allow customers to evaluate Cloud offerings and rank them based on their ability to meet the user's QoS requirements. Reference [48] has proposed a framework and a mechanism that measures the quality and prioritizes cloud services. The objective of [24] is to support the customers in reliably identifying trustworthy cloud providers. The objective of [49] is to

provide personalized trust management in which the user may play any of the three roles of consumer, broker, or provider. The objective of [50] is to provide decision making guidance to service providers to initialize collaborations by selecting trustworthy partners within the context of a cloud marketplace.

The objective of [51] is to provide a framework that enable trust-based cloud customer and cloud service provider interactions within the context of hybrid cloud computing environments. The objective of [27] is to employ a data-centric, detective approach to provide a framework (TrustCloud) to increase trust, security of data, and accountability in the Cloud at all levels of granularity. The aim of [34] is to provide usable confidentiality and integrity, through their Confidentiality as a Service (CaaS) paradigm for the majority of users for whom the current security mechanisms are too complex or require too much effort.

3.2.5 Billing and Accountability

The objective of [23] is to provide openness and transparency . They propose the notion of consumer–centric resource accounting model such that consumers can programmatically compute their consumption charges of a remotely used service. In particular, the notion of strongly consumer–centric accounting model is proposed that requires that all the data needed for calculating billing charges can be collected independently by the consumer (or a trusted third party, TTP).

According to [8], one of the major security obstacles to widespread adoption of cloud computing is the lack of near-real-time auditability. In particular, near-real-time cloud auditing, which provides timely evaluation results and rapid response, is the key to assuring the Cloud. Their objective is therefore to present strategies for reliable cloud auditing.

3.2.6 Concluding Remarks and Contexts of Studies

Usually, cloud providers provide assurances by specifying technical and functional descriptions in SLAs for the services they offer. The descriptions in SLAs are not consistent among the cloud providers even though they offer services with similar functionality. Customers are not sure whether they can identify a trustworthy cloud provider only based on its SLA. To support the customers in reliably identifying trustworthy cloud providers, [24] has proposed a multi-faceted trust management system architecture for a cloud computing marketplace. The context of [50] is the provision of guidance in the selection of trustworthy partners within a cloud computing marketplace. The context of [51] is to provide a framework that enable trust-based cloud customer and cloud service provider interactions within the context of hybrid cloud computing environments, where resource sharing between multiple Clouds to meet cloud user requirements are enabled by peering arrangements established between the participating Clouds. The context of [40] is the scheduling of resources of services in cloud computing environments by adopting a trust model based on Probabilistic Latent Semantic Analysis (pLSA) which analyzes the history information of each node and allocates reliable resources according to user requests.

Based on the findings from the above, the main arrears of trust engineering research focus has been on quality of service, security, access and identity management, user support on trust management, and accountability in in the context of a cloud computing marketplace . A major observation that I made from the reviewed studies is that the concept of trust is treated loosely without any formal specification or definition in the discussion of trust in cloud computing and trust engineering in general. Formal trust modeling and definitions are however very necessary in ensuring a unified view of the concept of trust in the design and engineering of trust management systems for cloud computing; this therefore deserves more attention from the cloud computing research community.

3.3 Modeling Trust

Reference [52] has carried out a survey on the trust management systems implemented on distributed systems with a special emphasis on cloud computing. They reported on several trust models such as CuboidTrust [53] , EigenTrust [54] , Bayesian Network based Trust Management (BNBTM) [55], GroupRep [56] , AntRep[57] , Global Trust[58] [59] , Peer Trust [60], and Trust Ant Colony System (TACS)[61]. These models were mainly proposed for systems like clusters, grids and wireless sensor networks, and have not been used or tested in cloud computing environments. Secondly, these models do not model the concept of trust but rather model practical trust management systems for distributed systems and their algorithms for acquiring and computing trust values.

This section is about the actual modeling of the concept of trust with a special focus on trust in cloud computing. We begin with looking at some definitions of trust and move on to obtaining a formalized model of the definition of the concept of trust in the context of cloud computing environments.

3.3.1 Definitions of Trust

Though there has been some work on trust modeling and trust management systems, and even in the new domain of trust management systems for cloud computing environments [62] [51] [63], the subjective nature of trust has made a solid definition elusive. Researchers have most often used the term loosely in their works; more specifically, a rigorous formal definition has not been applied in most cases. A few of the attempts at the definition of trust in the domain of trust engineering for cloud computing that was found during this study corroborates this observation. Salah and Eltoweissy [49] defined trust as the belief or

disbelief of a party that another party, for a said subject of trust, in a given context, has the intent, integrity, results and capability to exhibit a set of acceptable actions in the future, for the welfare of the trusting party. Viriyasitavat and Martin [64] has developed trust definition in the application domain of service workflows. They defined trust as "Trust is a subjective mutual measurable between interacting entities willing to act dependably, securely, and reliably, in a given situation within specific context of a given time". Their definition is an adaptation of that of Olmedilla, et al [65] which states that "Trust of a party A to a party B for a service X is the measurable belief of A in that B behaves dependably for a specified period within a specified context (in relation to service X)". It should be noted that whilst in the domain of service workflows, being able to establish trust in both directions is crucial, as one service may need to verify trustworthiness of a subsequent service before passing information, and the subsequent service perhaps requires trust that an outcome must be originated from a trusted source, the definition of Viriyasitavat & Martin contradicts the generally accepted asymmetry property of trust relationships.

Dellarocas' definition of trust [66] is adopted in this work. Its salient points are summarized below and explained in the context of cloud computing.

The level of trust $T_c^s(t_i)$ of a service consumer c for a service provider s in the context of a transaction $t_i \in T$ is the a priori probability that the utility of c will meet or exceed its minimum threshold of satisfaction u_0 at the end of transaction t_i, given $c's$ perceived trustworthiness of service provider s. Simply stated, trust is the level of confidence of c that the outcome of a transaction with another agent s will be satisfactory for it. More formally: $T_c^s(t_i) = \int_{U_c(R) \geq u_0} \tau_c^s(R, t_i).dR$, where $U_c(R)$ is the utility function of service consumer c; and $\tau_c^s(R, t_i)$ - the trustworthiness of service provider s as perceived by consumer c in the context of a transaction $t_i \in T$ is the a priori subjective joint probability distribution function of the critical rating vector $R_c^s(t_i)$ from the perspective of c.

It is not only cloud service consumers that need the consideration of trust in their transactions with the cloud service providers. Most often than not, cloud services providers also need to be wary of the activities of cloud service consumers. Thus, trust modeling is useful in the analysis of the genuine and potentially malicious service consumers. Therefore a trust model is needful for the perceived trustworthiness of service consumers by the providers of the services. So similarly, the level of trust $T_s^c(t_i)$ of a service provider s for a service consumer c in the context of a transaction $t_i \in T$ is the a priori probability that the utility of s will meet or exceed its minimum threshold of satisfaction u_0 at the end of transaction t_i, given service provider s perceived trustworthiness of service consumer c. Again, more formally: $T_s^c(t_i) = \int_{U_s(R) \geq u_0} \tau_s^c(R, t_i).dR$, where $U_s(R)$ is the utility function of service provider s; and $\tau_s^c(R, t_i)$ - the trustworthiness of service consumer c as perceived by service provider s in the context of a transaction $t_i \in T$ is the a priori subjective joint probability distribution function of the critical rating vector $R_s^c(t_i)$ from the perspective of s. Please note that it is for notational simplicity that the critical rating vectors $R_c^s(t_i)$ and $R_s^c(t_i)$ are denoted by R (without the full complement of the subscripts) in the denotation of the trustworthiness.

The above definitions have a number of interesting properties which correspond with the intuitive properties of trust in our everyday life such as trustworthiness is subjective, and it is defined relative to a particular set of critical attributes; trustworthiness is defined at a given point in time, and it is defined as a probability distribution. Some other important intuitive attributes of trust are that trust has duality - it is subjective and objective; that is, some of the critical attributes are subjectively measureable and others are objectively measureable; trust is not always symmetrical; and trust is dynamic, that is, trust is related to environment (context) and temporal factors [67].

3.3.2 Cloud Computing Parameters of Trust

When selecting a cloud service provider, multiple important parameters that are of relevance to the cloud service consumer need to be identified properly. Also, there is need for mechanisms to measure those parameters and aggregate these measurements based on the customers' preference regarding the importance of the parameters[68]. Ref. [69] and [68] have identified several of these parameters which have been categorized into quality of service related, security and privacy related, risk management related, and reputation related attributes. These parameters (attributes) are termed critical attributes; more formerly, a *critical attribute* of a service provider s, from the perspective of a service consumer c, in the context of a transaction $t_i \in T$ is an attribute whose value affects the utility of c and is contingent upon the behavior of s in the course of transaction t_i [66]. A non-exhaustive list of selected set of the potential critical attributes in cloud services are briefly outlined below under each of these categories.

3.3.2.1 Quality of Service Related Attributes

International Telecommunication Union has defined a methodology for capturing the quality requirements of a user of communication services which uses seven general criteria [70]. This view is modified in [71] by adding capability, usability, and fidelity - as a supplement to accuracy. Each of these generic aspects can be applied at different stages of the SLA lifecycle, and are applicable to cloud services. They therefore remain useful dimensions along which to classify cloud services [72]. The QoS related elements are performance metrics such as latency, availability, accuracy, reliability, and capability [72]. These metrics have also been emphasized by [48] and also asserted by [73] to be part of their ten common denominators that must be considered to make cloud storage valuable.

3.3.2.2 Security and Privacy Related Attributes

Some of the security and privacy related parameters that are pertinent to cloud consumers and cloud service providers are data confidentiality and integrity, federated identity management solutions, secure authentication and session management, and secure cryptographic mechanisms. Other prevalent vulnerabilities in state-of-the-art cloud computing offerings that cloud consumers are wary of include SQL injection, command injection and cross-site scripting. Some of the security parameters that are more pertinent to cloud services providers are key management, physical security support, network security support, unauthorized access to management interface, and internet protocol vulnerabilities.

3.3.2.3 Risk Management Related Attributes

Some of the risk management related factors that are of importance to cloud consumers are standardized SLA with unambiguous guarantees, near-real time auditing services [8] and visibility into the security controls and processes employed by the service provider as well as their performance over time that offer transparency, compliance (accreditation or certification), security measures, interoperability, customer support facilities, geographical location of cloud storage (data protection laws and other jurisdictional implication of where data is stored), and cloud service deployment models.

3.3.2.4 Reputation Related Attributes

Reputation related parameters form some of the potential critical attributes that users consider in selecting cloud services. Some of these parameters are recommendation from existing users of the service, feedback and publicly available reviews of the specific cloud services, category of the service and reputation of the cloud service provider.

3.3.2.5 General Cloud Metrics of Trust

In addition to the cloud specific attributes, some general attributes that are dependent on the activities of an entity to be trusted are of relevance for our discussion. The four main attributes of this category are intent, integrity, capability and results. Intent constitutes information about declared agendas (what parties promise to provide through their services), integrity constitutes information about honesty (if parties deliver what they promised), capability constitutes information about owned or outsourced resources, and results constitute information about products they are specialized in [49].

3.4 Trust Engineering Approaches

The various major approaches towards trust engineering in cloud computing is presented in this section. It should be evident to readers that any research work that targets one or more of the trust attributes (or other related trust attributes) discussed in Section 3.3.2 above contributes to trust engineering in cloud computing. We identify two broad categories based on whether it is targeted towards benefiting cloud service consumers or the cloud service providers. The identified major approaches to trust engineering in cloud computing are cloud audit based, reputation based, trusted third party based, trusted computing technology based, and cloud services deployment based approaches.

3.4.1 End User Support Oriented Trust Engineering

This is about mechanisms that facilitate building cloud consumers' trust in choosing and managing cloud service usage.

3.4.1.1 Cloud Audit Approaches

Reference [23] has proposed the notion of a Consumer–centric Resource Accounting Model for a cloud resource. An accounting model is weakly consumer-centric if all the data that the model requires for calculating billing charges can be queried programmatically from the provider. Further, an accounting model is strongly consumer–centric if all the data that the model requires for calculating billing charges can be collected independently by the consumer (or a TTP); in effect, this means that a consumer (or a TTP) should be in a position to run their own measurement service. They contend that it is in the interest of the providers to make their accounting models at least weakly consumer-centric. Strongly consumer–centric models should prove even more attractive to consumers as they enable consumers to incorporate independent consistency or reasonable checks as well as raise alarms when apparent discrepancies are suspected in consumption figures. Strongly consumer-centric accounting models have the desirable property of openness and transparency, since service users are in a position to verify the charges billed to them.

One of the most common groupings or layers in cloud computing is the view of IaaS, PaaS and SaaS. These abstractions layers are mainly system-centric. In contrast, the *TrustCloud* framework takes a different perspective, i.e., an architectural, data-centric view. Because of the

scale of cloud computing, the types of data-centric logs range from system-level file-centric logs to workflow-level audit trail logs. The *TrustCloud* framework attempts to describe the layers of cloud accountability. The five abstraction layers of the types of logs needed for an accountable cloud are system layer – addresses tracking of files across the Cloud, data layer – addresses tracking of change of data and information across the Cloud, workflow layer – addresses data and information flow in the Cloud, law and regulations layer – addresses data-centric logging requirements mandated by external laws and regulations, and finally, policies layer – addresses data-centric audit requirements mandated by internal governance and audit requirements [27].

3.4.1.2 Reputation Based Approaches

Reference [51] presents a fully distributed framework that enable trust-based cloud customer and cloud service provider interactions. The framework aids a service consumer in assigning an appropriate weight to the feedback of different raters regarding a prospective service provider. They developed a mechanism based on their framework for controlling falsified feedback ratings from iteratively exerting trust level contamination due to falsified feedback ratings.

Secure integrity attestation of computation results is the focus of [29]. Whereas AdapTest [74] and RunTest [75] implement cloud service integrity attestation for the IBM ***System S*** stream processing system [76] using attestation graphs in which always-agreeing nodes form a clique in the graph, facilitating detection of malicious collectives; in contrast, the work of [29] considers a reputation-based trust management approach to integrity violation detection in Hadoop clouds. Trust management systems probabilistically anticipate future misbehavior of untrusted agents based on their histories of past behavior.

3.4.1.3 Trusted Third Party Based Approaches

The goal of [43] is the remote assessment of the cloud infrastructure's integrity by a cloud certifier. They hence need to detect all changes in the remote system that can possibly compromise security. All changes in the hardware or software should be reported to the cloud certifier, even if the infrastructure provider has super-user access to the machine. Their *BonaFides* system monitors the infrastructure provider's physical hosts by observing file modifications on a low level and persistently stores the history of these integrity measurements and file changes. Files are measured at regular intervals and whenever changes in the files are detected. *BonaFides* measures the hypervisor, kernel, kernel modules, disk and network utilities, and system configuration files in the Dom0 (the administrative domain of the Xen hypervisor that manages access to the physical host's resources).

3.4.1.4 Trusted Computing Technology Base Approaches

Ref. [77] has presented a multi-tenancy trusted computing environment model (MTCEM) to support the security duty separation between Cloud Service Provider (CSP) and customers. MTCEM is designed for IaaS service delivery model, and it intends to separate the security responsibility of the CSP and their customers on cloud infrastructures. In MTCEM model, CSP is responsible to assure a trusted host and Virtual Machine Monitor (VMM) environment, and customers are responsible for the assurance of trusted virtual instances they rent from CSP. MTCEM uses the two main mechanisms of transitive trust and platform attestation of the trusted computing technology. It uses transitive trust mechanism to build a trusted computing platform and attestation mechanism to improve the customers' confidence on CSP. Ref. [25] shows how to design the Trusted SaaS Platform (TSP) by taking advantage of trusted computing technologies. Conventional trusted computing platforms like Terra [78] are able to prevent the owner of a physical machine from inspecting or interfering with a computation running in a virtual machine (VM) that is hosted in the physical machine, and thus can effectively secure the computation running in the VM. However, these platforms cannot address security and trust issues in SaaS environments due to the following two reasons. First, they do not specify who will launch the VM that is responsible for performing the computation. The approach presented in Towards Trusted Cloud Computing [79] on Trusted Cloud Computing Platform (TCCP) can only be used for IaaS and not suitable for SaaS environments. In TCCP, the protocols are mainly utilized for node registration and securing VM launch and migration. However, in SaaS system, the users' main purpose is guaranteeing that the SaaS providers process their data and respond with the result without inspection or modification, rather than guaranteeing the security of their VMs. To address this problem, [25] proposed a trusted SaaS platform that enables a trusted third party to launch a VM as a trusted execution environment(TEE) on the computation server. Thus though the privileged administrators of SaaS providers can access the physical host of TEE, they cannot access the TEE because the TEE is not launched by them. The TSP leverages the trusted virtual machine monitor (TVMM) [78] so privileged administrators cannot tamper with the TEE. The TEE is also where all of the decryption, computation and encryption take place, so it can ensure the confidentiality and integrity of users' data and computations outsourced to SaaS services.

3.4.1.5 Cloud Service Deployment Approaches

Reference [69] has devised five reference deployment models for cloud computing that progressively address user security concerns and increase users' trust in cloud computing. These are the separation model, availability model, migration model, tunnel model, and encryption

model. The Separation Model is the base model for all the other four models. It separates data storage from data processing, requiring at least two independent cloud service providers to process data and to store data, respectively. This can help ease users' concerns on having a single provider in complete control over the data and the services they use. The Availability Model introduces redundancy into the Separation Model, in both the data processing and the data storage. With the redundancy in the Availability Model, failures of one data processing service and one data storage service can be tolerated. The Tunnel Model further enhances the Separation Model by using a Tunnel Service to impose isolation between the Data Processing Service and the Cloud Storage Service. The Tunnel Service prevents collusion by cutting the direct communications between the Data Processing Service and the Cloud Storage Service, assuming that it is very unlikely for two isolated providers to collude. The Cryptography Model augments the Tunnel Model with cryptography support, such as data encryption, decryption, and digital signing.

Even though there are approaches to provide confidentiality for the users' data in the Cloud, these are not widely adopted due to both awareness and usability issues. Therefore, [34] proposed the Confidentiality as a Service (CaaS) paradigm to provide usable confidentiality and integrity for the bulk of users for whom the current security mechanisms are too complex or require too much effort. The CaaS paradigm combines data security with usability by design and integrates effortlessly into available cloud service applications and workflows. They leverage the splitting of trust between the cloud service provider and one or more CaaS providers to improve usability. CaaS focuses on unobtrusive confidentiality by hiding all cryptographic artifacts from the prevalently non-technical users [34].

3.4.2 Service Provider Oriented Trust Engineering

This facilitates building trust between the cloud service providers and their customers in ensuring that their resources and administrative platforms will not be abused by the consumers.

3.4.2.1 Reputation Based Approaches

Reference [50] considers the scenario where a service provider, termed the Master Service Provider (MSP), identifies a great business opportunity or other scenarios which need collaboration with other service providers, termed Guest Service Providers (GSP), to offer a set of new services to the customers. Their approach is to derive trustworthiness of guest service provider *i* (GSP*i*) according to its past behavior.

3.4.2.2 Identity and Access Management

Identity management is one of the core components in cloud privacy and security and can help alleviate some of the user trust issues associated with cloud computing. Available solutions use trusted third party in identifying entities to service providers. The solution providers do not recommend the usage of their solutions on untrusted hosts. Ref. [33] has proposed an approach for identity management that is independent of trusted third parties and has the ability to use identity data on untrusted hosts. The approach is based on the use of predicates over encrypted data and multi-party computing for negotiating a use of a cloud service. It uses active bundle - which is a middleware agent that includes PII, privacy policies, a virtual machine that enforces the policies, and has a set of protection mechanisms to protect it. An active bundle interacts on behalf of a user to authenticate to cloud services using user's privacy policies.

3.4.3 Final Remarks

Ref. [80] has argued that cryptography alone can't enforce the privacy demanded by common cloud computing services, even with such powerful tools as fully homomorphic encryption (FHE). They formally define a hierarchy of natural classes of private cloud applications, and show that no cryptographic protocol can implement those classes where data is shared among clients.

Employing trusted computing technologies and reputation based approaches are two key approaches to trust engineering in cloud computing marketplace. Also the adopted cloud deployment model plays a significant role in improving trust in cloud environments.

3.5 Trust Management Systems

Trust management is the activity of creating systems and methods that allow relying parties to make assessments and decisions regarding the dependability of potential transactions involving risk, and that also allow players and system owners to increase and correctly represent the reliability of themselves and their systems [81]. There is a need for methodologies that enable relying parties to determine the trustworthiness of remote parties through computer mediated communication and collaboration. At the same time, trustworthy entities need methodologies that enable them to be recognized as such; developing and applying these methodologies can be called trust management.

A survey on the trust management systems implemented on distributed systems with emphasis on cloud computing has been carried out by [52] . They reported on several trust models such as *CuboidTrust* [53] , *EigenTrust*[54] , *Bayesian Network based Trust Management (BNBTM)* [55], *GroupRep*[56] , *AntRep*[57] , *Global Trust*[58] [59] , *Peer Trust* [60], *and Trust Ant Colony System (TACS)*[61]. These models were mainly proposed for systems like clusters, grids and wireless sensor networks, and have not been used or tested in cloud computing environments; hence their suitability for use in cloud computing cannot be recommended without an extensive evaluation. Though a few work on trust

targeting cloud computing environments were considered in [52], it was found that none of the proposed systems was based on solid theoretical foundation and also does not take any quality of service attribute into account for forming the trust scores. This observation may be due in part to the fact that, although the considered studies dealt with elements of trust in cloud computing and hence will pass for approaches to trust engineering in cloud computing, these were not really trust management systems since they do not possess elements for the generic operations of trust management systems which include expectation, data monitoring, data management, analysis, and decision making. Secondly, this observation is also partly due to the fact that the concept of trust itself is still not well understood by the research community due to its loose usage without formal specification. Hence a solid formulation of the concept of trust is essential for the research community, and more especially in the context of cloud computing in order to lay solid theoretical foundation for building trust management systems for cloud computing.

Some of the trust related works in cloud computing that have provided some generic methodologies in developing trust management systems for cloud computing environments are [49] and [24]. The generic operations of trust management include expectation, data monitoring, data management, analysis, and decision making. Separation of these operations supports data privacy, confidentially and integrity, where data can be kept at their sources and accessed only on a need to know basis[49]. The model builds trust using the four parameters: intent, integrity, capability and results. Intent constitutes information about declared agendas about what entities promise to provide through their services. Integrity constitutes information about honesty which is a measure of, to what extent entities deliver on what they promised. Capability constitutes information about owned or outsourced resources; and finally, results constitute information about products and services that entities specialized in through consistently delivering these products and services satisfactorily to their clients.

3.5.1 Final Remarks on Trust Management Systems

The current state-of-the-art in trust management systems are that, they are mainly for peer-to-peer systems. Secondly, current trust systems provide no separation of concern among different trust management operations. Also most current trust management systems provide limited or no customization according to trusting entities' requirements. The focus is skewed towards service providers being evaluated by service consumers for their trustworthiness, but not vice versa[49]. In addition to designing trust management systems that factor in the above mentioned points, the solid formulation of the concept of trust is essential for the research community, and more especially in the context of cloud computing in order to lay solid theoretical foundation for building trust systems for cloud computing environments.

4 CONCLUSION & FUTURE WORK

This work has reviewed identified primary studies on trust engineering approaches in cloud computing. The central motivating objective of this work has been to lay the foundation for designing a trust management system for OCS platforms, and provide summary of trust engineering approaches in cloud computing for easy reference by the research community. The study has been specifically interested in finding the main approaches towards trust engineering in cloud computing, the objectives of the identified primary studies and in what contexts these trust management systems are being developed; and finally, the major trust models and trust management systems for cloud computing.

It was observed that trusted computing technologies and reputation based approaches are the main approaches to trust engineering in cloud computing. Also trusted third party approaches and the deployment model play a significant role in enhancing trust between service providers and consumers.

Based on the findings during the study, the main arrears of trust engineering research focus has been on quality of service, security, access and identity management, user support on trust management, and accountability in in the context of a cloud computing marketplace .

We observed that the concept of trust is used loosely without any formal specification in cloud computing discussions and trust engineering in general. As a first step towards addressing this problem, we have contextualized the formal trust specification in multi-agent environments for cloud computing. This should prove very useful for other researchers interested in trust related research in a cloud computing marketplace.

The findings in this paper have been applied in the design of a trust management system for opportunistic cloud services [82]. We will as part of our future work, expand on the concept of composite (group) trust, and provide suitable formal specification and definition for it.

5 LIMITATIONS OF STUDY

There could be a possible bias of the authors during the practical screening process towards selecting relevant primary studies based on personal interest in studies that are based on concepts similar to that of opportunistic cloud services. This is because since the central motivating objective of this work is to lay the foundation for designing a trust management system for opportunistic cloud services platforms, studies that have elements of concepts similar to that this are of interest to the authors. With this concern in mind from the

beginning of this work, deliberate steps were however taken to ensure that this inherent bias does not affect the selection of the included primary studies.

REFERENCES

[1] E. Kuada and H. Olesen, "A Social Network Approach to Provisioning and Management of Cloud Computing Services for Enterprises," in *Proceedings of The Second International Conference on Cloud Computing, GRIDs, and Virtualization*, 2011, pp. 98–104.

[2] E. Kuada, H. Olesen, and A. Henten, "Public Policy and Regulatory Implications for the Implementation of Opportunistic Cloud Computing Services for Enterprises," in *9th International Workshop on Security in Information Systems*, Wroclaw, Poland, 2012, pp. 3 – 13.

[3] W. Jansen and T. Grance, "Guidelines on security and privacy in public cloud computing," *NIST Special Publication*, pp. 800–144, 2011.

[4] L. Badger, T. Grance, R. Patt-Corner, and J. Voas, "Draft cloud computing synopsis and recommendations," *NIST Special Publication*, vol. 800, p. 146, 2011.

[5] P. Mell and T. Grance, "The NIST definition of cloud computing (draft)," *NIST special publication*, vol. 800, p. 145, 2011.

[6] Z. Song, J. Molina, and C. Strong, "Trusted Anonymous Execution: A Model to Raise Trust in Cloud," in *2010 9th International Conference on Grid and Cooperative Computing (GCC)*, 2010, pp. 133 – 138.

[7] M. Almorsy, J. Grundy, and A. S. Ibrahim, "TOSSMA: A Tenant-Oriented SaaS Security Management Architecture," in *2012 IEEE 5th International Conference on Cloud Computing (CLOUD)*, 2012, pp. 981 –988.

[8] J. S. Park, E. Spetka, H. Rasheed, P. Ratazzi, and K. J. Han, "Near-Real-Time Cloud Auditing for Rapid Response," in *2012 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, 2012, pp. 1252 –1257.

[9] E. Kuada and H. Olesen, "Incentive mechanisms for Opportunistic Cloud Computing Services," in *2012 8th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, Pittsburgh, PA, USA, 2012, pp. 127 –136.

[10] E. Kuada, K. Adanu, and H. Olesen, "Cloud Computing and Information Technology Resource Cost Management for SMEs," in *Proceedings of IEEE Region 8 Conference EuroCon 2013*, University of Zagreb, Croatia, 2013, pp. 258 – 265.

[11] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 2007.

[12] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of Systems and Software*, vol. 80, no. 4, pp. 571–583, Apr. 2007.

[13] C. Okoli and K. Schabram, "A Guide to Conducting a Systematic Literature Review of Information Systems Research," in *Working Papers on Information Systems*, 2010.

[14] Y. Levy and T. J. Ellis, "A Systems Approach to Conduct an Effective Literature Review in Support of," *INFORMATION SYSTEMS RESEARCH. INFORMING SCIENCE JOURNAL*, vol. 9, p. 181212, 2006.

[15] L. M. Vaquero, L. Rodero-Merino, and D. Morán, "Locking the sky: a survey on IaaS cloud security," *Computing*, vol. 91, no. 1, pp. 93–118, Jan. 2011.

[16] C. Modi, D. Patel, B. Borisaniya, A. Patel, and M. Rajarajan, "A survey on security issues and solutions at different layers of Cloud computing," *J Supercomput*, pp. 1–32, Oct. 2012.

[17] K. Hashizume, D. G. Rosado, E. Fernández-Medina, and E. B. Fernandez, "An analysis of security issues for cloud computing," *Journal of Internet Services and Applications*, vol. 4, no. 1, p. 5, Feb. 2013.

[18] RefWorks, "RefWorks," 2013. [Online]. Available: http://www.refworks.com/.

[19] nVivo10, "Nvivo," nVivo10. [Online]. Available: http://www.qsrinternational.com/products_nvivo.aspx.

[20] S. Friese, "ATLAS.ti 7 User Guide and Reference: ATLAS.ti 7 USER MANUAL." ATLAS.ti Scientific Software Development GmbH, Berlin, 28-Jan-2013.

[21] K. Charmaz, "Grounded theory," *Strategies of qualitative inquiry*, vol. 2, p. 249, 2003.

[22] A. Strauss and J. Corbin, "Grounded theory methodology," *Handbook of qualitative research*, pp. 273–285, 1994.

[23] A. Mihoob, C. Molina-Jimenez, and S. Shrivastava, "A Case for Consumer-centric Resource Accounting Models," in *2010 IEEE 3rd International Conference on Cloud Computing (CLOUD)*, 2010, pp. 506 –512.

[24] S. M. Habib, S. Ries, and M. Muhlhauser, "Towards a Trust Management System for Cloud Computing," in *2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2011, pp. 933 –939.

[25] C. Zhong, J. Zhang, Y. Xia, and H. Yu, "Construction of a Trusted SaaS Platform," in *2010 Fifth IEEE International Symposium on Service Oriented System Engineering (SOSE)*, 2010, pp. 244 –251.

[26] M. Almorsy, J. Grundy, and A. S. Ibrahim, "Collaboration-Based Cloud Computing Security Management Framework," in *2011 IEEE International Conference on Cloud Computing (CLOUD)*, 2011, pp. 364 –371.

[27] R. K. L. Ko, M. Kirchberg, and B. S. Lee, "From system-centric to data-centric logging - Accountability, trust amp; security in cloud computing," in *Defense Science Research Conference and Expo (DSR), 2011*, 2011, pp. 1 –4.

[28] A. Hadoop, *Apache Hadoop*. 2013.

[29] S. M. Khan and K. W. Hamlen, "Hatman: Intra-cloud Trust Management for Hadoop," in *2012 IEEE 5th International Conference on Cloud Computing (CLOUD)*, 2012, pp. 494 –501.

[30] B. Panzer-Steindel, "Data integrity," CERN/IT, Draft Draft 1.3, Apr. 2007.

[31] S. Narayan, J. A. Chandy, S. Lang, P. Carns, and R. Ross, "Uncovering errors: the cost of detecting silent

data corruption," in *Proceedings of the 4th Annual Workshop on Petascale Data Storage*, New York, NY, USA, 2009, pp. 37–41.

[32] S. Nepal, S. Chen, J. Yao, and D. Thilakanathan, "DIaaS: Data Integrity as a Service in the Cloud," in *2011 IEEE International Conference on Cloud Computing (CLOUD)*, 2011, pp. 308 –315.

[33] R. Ranchal, B. Bhargava, L. B. Othmane, L. Lilien, A. Kim, M. Kang, and M. Linderman, "Protection of Identity Information in Cloud Computing without Trusted Third Party," in *2010 29th IEEE Symposium on Reliable Distributed Systems*, 2010, pp. 368 –372.

[34] S. Fahl, M. Harbach, T. Muders, and M. Smith, "Confidentiality as a Service – Usable Security for the Cloud," in *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2012, pp. 153 –162.

[35] M. M. Lucas and N. Borisov, "FlyByNight: mitigating the privacy risks of social networking," in *Proceedings of the 7th ACM workshop on Privacy in the electronic society*, New York, NY, USA, 2008, pp. 1–8.

[36] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: an online social network with user-defined privacy," in *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, New York, NY, USA, 2009, pp. 135–146.

[37] A. Whitten and J. D. Tygar, "Why Johnny can't encrypt: a usability evaluation of PGP 5.0," in *Proceedings of the 8th conference on USENIX Security Symposium - Volume 8*, Berkeley, CA, USA, 1999, pp. 14–14.

[38] P. Gutmann, "Plug-and-play PKI: a PKI your mother can use," in *Proceedings of the 12th conference on USENIX Security Symposium - Volume 12*, Berkeley, CA, USA, 2003, pp. 4–4.

[39] D. Balfanz, G. Durfee, D. K. Smetters, and R. E. Grinter, "In search of usable security: five lessons from the field," *IEEE Security Privacy*, vol. 2, no. 5, pp. 19 – 24, Oct. 2004.

[40] H. Kim, H. Lee, W. Kim, and Y. Kim, "A Trust Evaluation Model for QoS Guarantee in Cloud Systems," *International Journal of Grid and Distributed Computing*, vol. 3, no. 1, pp. 1 – 10, Mar. 2010.

[41] íñIgo Goiri, F. Julií, J. O. Fitó, M. MacíAs, and J. Guitart, "Supporting CPU-based guarantees in cloud SLAs via resource-level QoS metrics," *Future Gener. Comput. Syst.*, vol. 28, no. 8, pp. 1295–1302, Oct. 2012.

[42] X. Liu, Y. Yang, D. Yuan, G. Zhang, W. Li, and D. Cao, "A Generic QoS Framework for Cloud Workflow Systems," in *Proceedings of the 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, Washington, DC, USA, 2011, pp. 713–720.

[43] R. Neisse, D. Holling, and A. Pretschner, "Implementing Trust in Cloud Infrastructures," in *2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 2011, pp. 524 – 533.

[44] F. Doelitzscher, C. Reich, and A. Sulistio, "Designing Cloud Services Adhering to Government Privacy Laws," in *2010 IEEE 10th International Conference on Computer and Information Technology (CIT)*, 2010, pp. 930 –935.

[45] "Federal Information Security Management Act (FISMA)," 2002. [Online]. Available: http://www.dhs.gov/federal-information-security-management-act-fisma. [Accessed: 27-Feb-2013].

[46] G. Stoneburner, A. Y. Goguen, and A. Feringa, "SP 800-30. Risk Management Guide for Information Technology Systems," National Institute of Standards & Technology, Gaithersburg, MD, United States, 2002.

[47] M. Ahmed and Y. Xiang, "Trust Ticket Deployment: A Notion of a Data Owner's Trust in Cloud Computing," in *2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2011, pp. 111 –117.

[48] S. Garg, S. Versteeg, and R. Buyya, "A framework for ranking of cloud computing services," *Future Generation Computer Systems*, Jun. 2012.

[49] H. Salah and M. Eltoweissy, "Towards a personalized trust management system," in *2012 International Conference on Innovations in Information Technology (IIT)*, 2012, pp. 373 –378.

[50] L. Xin and A. Datta, "On trust guided collaboration among cloud service providers," in *2010 6th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, 2010, pp. 1 –8.

[51] J. Abawajy, "Establishing Trust in Hybrid Cloud Computing Environments," in *2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2011, pp. 118 –125.

[52] M. Firdhous, O. Ghazali, and S. Hassan, "Trust Management in Cloud Computing: A Critical Review," *International Journal on Advances in ICT for Emerging Regions (ICTer)*, vol. 4, no. 2, pp. 24–36, 2012.

[53] R. Chen, X. Zhao, L. Tang, J. Hu, and Z. Chen, "CuboidTrust: a global reputation-based trust model in peer-to-peer networks," *Autonomic and Trusted Computing*, pp. 203–215, 2007.

[54] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 640–651.

[55] Y. Wang, V. Cahill, E. Gray, C. Harris, and L. Liao, "Bayesian network based trust management," *Autonomic and Trusted Computing*, pp. 246–257, 2006.

[56] H. Tian, S. Zou, W. Wang, and S. Cheng, "A group based reputation system for P2P networks," *Autonomic and trusted computing*, pp. 342–351, 2006.

[57] W. Wang, G. Zeng, and L. Yuan, "Ant-based reputation evidence distribution in P2P networks," in *Grid and Cooperative Computing, 2006. GCC 2006. Fifth International Conference*, 2006, pp. 129–132.

[58] F. Yu, H. Zhang, F. Yan, and S. Gao, "An improved global trust value computing method in P2P system," *Autonomic and trusted computing*, pp. 258–267, 2006.

[59] W. Wang, X. Wang, S. Pan, and P. Liang, "A new global trust model based on recommendation for peer-to-peer network," in *New Trends in Information and*

Service Science, 2009. NISS'09. International Conference on, 2009, pp. 325–328.

[60] L. Xiong and L. Liu, "PeerTrust: supporting reputation-based trust for peer-to-peer electronic communities," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843 – 857, Jul. 2004.

[61] F. Gómez Mármol, G. Martínez Pérez, and A. F. Gómez Skarmeta, "TACS, a trust model for P2P networks," *Wireless personal communications*, vol. 51, no. 1, pp. 153–164, 2009.

[62] X. Zhang, H. Liu, B. Li, X. Wang, H. Chen, and S. Wu, "Application-Oriented Remote Verification Trust Model in Cloud Computing," in *2010 IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom)*, 2010, pp. 405 – 408.

[63] M. Kuehnhausen, V. S. Frost, and G. J. Minden, "Framework for assessing the trustworthiness of cloud resources," in *2012 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, 2012, pp. 142 –145.

[64] W. Viriyasitavat and A. Martin, "Formal Trust Specification in Service Workflows," in *2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, 2010, pp. 703 –710.

[65] D. Olmedilla, O. F. Rana, B. Matthews, and W. Nejdl, "Security and trust issues in semantic grids," *In Proceedings of the Dagstuhl Seminar, Semantic Grid: the convengence of technologies, Volume 05271. 2005. [PD05] [PPI04] Panteli*, pp. 191–200, 2005.

[66] C. Dellarocas, "The Design of Reliable Trust Management Systems for Electronic Trading Communities," in *SLOAN SCHOOL OF MANAGEMENT, MIT, 2000*, 2001.

[67] C. Shen, H. Zhang, H. Wang, J. Wang, B. Zhao, F. Yan, F. Yu, L. Zhang, and M. Xu, "Research on trusted computing and its development," *Sci. China Inf. Sci.*, vol. 53, no. 3, pp. 405–433, Mar. 2010.

[68] S. M. Habib, S. Ries, and M. Muhlhauser, "Cloud Computing Landscape and Research Challenges Regarding Trust and Reputation," in *2010 7th International Conference on Ubiquitous Intelligence Computing and 7th International Conference on Autonomic Trusted Computing (UIC/ATC)*, 2010, pp. 410 –415.

[69] G. Zhao, C. Rong, M. G. Jaatun, and F. E. Sandnes, "Reference deployment models for eliminating user concerns on cloud security," *J Supercomput*, vol. 61, no. 2, pp. 337–352, Aug. 2012.

[70] "Communications quality of service: A framework and definitions," International Telecommunication Union, Recommendation ITU-T Recommendation G.1000, Nov. 2001.

[71] ETSI, "User Group; Quality of telecom services; Part 1: Methodology for identification of parameters relevant to the Users," European Telecommunications Standards Institute, Sophia Antipolis Cedex - FRANCE, Guide ETSI EG 202 009-1 V1.2.1 (2006-11), Nov. 2006.

[72] "CLOUD; SLAs for Cloud services," European Telecommunications Standards Institute, 650 Route des Lucioles F-06921 Sophia Antipolis Cedex - FRANCE, Technical Report ETSI TR 103 125 V1.1.1 (2012-11), Nov. 2012.

[73] PROMISE Technology Inc., "Cloud Computing andTrusted Storage." PROMISE Technology Inc., Q1-2010.

[74] J. Du, N. Shah, and X. Gu, "Adaptive data-driven service integrity attestation for multi-tenant cloud systems," in *2011 IEEE 19th International Workshop on Quality of Service (IWQoS)*, 2011, pp. 1 –9.

[75] J. Du, W. Wei, X. Gu, and T. Yu, "RunTest: assuring integrity of dataflow processing in cloud computing infrastructures," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, New York, NY, USA, 2010, pp. 293–304.

[76] B. Gedik, H. Andrade, K.-L. Wu, P. S. Yu, and M. Doo, "SPADE: the system s declarative stream processing engine," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, New York, NY, USA, 2008, pp. 1123–1134.

[77] X.-Y. Li, L.-T. Zhou, Y. Shi, and Y. Guo, "A trusted computing environment model in cloud architecture," in *2010 International Conference on Machine Learning and Cybernetics (ICMLC)*, 2010, vol. 6, pp. 2843 – 2848.

[78] T. Garfinkel, B. Pfaff, J. Chow, M. Rosenblum, and D. Boneh, "Terra: a virtual machine-based platform for trusted computing," 2003, pp. 193–206.

[79] N. Santos, K. P. Gummadi, and R. Rodrigues, "Towards Trusted Cloud Computing," in *HOTCLOUD*, 2009.

[80] M. Van Dijk and A. Juels, "On the impossibility of cryptography alone for privacy-preserving cloud computing," in *Proceedings of the 5th USENIX conference on Hot topics in security*, Berkeley, CA, USA, 2010, pp. 1–8.

[81] A. Jøsang, C. Keser, and T. Dimitrakos, "Can we manage trust?," in *Proceedings of the Third international conference on Trust Management*, Berlin, Heidelberg, 2005, pp. 93–107.

[82] E. Kuada, "Trust Management System for Opportunistic Cloud Services," in *2013 IEEE 2nd International Conference on Cloud Networking (CloudNet)*, San Francisco, USA, 2013, pp. 33 – 41.